

Sequence Analysis using Logic Regression

Charles Kooperberg Ingo Ruczinski, Michael L. LeBlanc, and Li Hsu

Division of Public Health Sciences, Fred Hutchinson Cancer Research Center, Seattle, Washington

Logic Regression is a new adaptive regression methodology that attempts to construct predictors as Boolean combinations of (binary) covariates. In this paper we modify this algorithm to deal with single-nucleotide polymorphism (SNP) data. The predictors that are found are interpretable as risk factors of the disease. Significance of these risk factors is assessed using techniques like cross-validation, permutation tests, and independent test sets. These model selection techniques remain valid when data is dependent, as is the case for the family data used here. In our analysis of the Genetic Analysis Workshop 12 data we identify the exact locations of mutations on gene 1 and gene 6 and a number of mutations on gene 2 that are associated with the affected status, without selecting any false positives.

Key words: adaptive estimation, Boolean combinations, simulated annealing, SNP

LOGIC REGRESSION

Finding associations between many genes/environmental factors and disease outcomes leads to statistical problems with a high-dimensional predictor space. In this paper we first discuss a new adaptive regression methodology, Logic Regression, which we apply to sequence data for the general population of the Genetic Analysis Workshop (GAW) 12 data. Logic Regression [Ruczinski 2000; Ruczinski et al., 2001] is intended for situations where most predictors are binary (0/1), and the goal is to find Boolean combinations of these predictors that are associated with an outcome variable. First assume that all predictors X_i , $i = 1, \dots, p$ are binary and write X_i instead of $\text{Ind}(X_i = 1)$ and X_i^c instead of $\text{Ind}(X_i = 0)$, where $\text{Ind}(\cdot)$ is the usual indicator function. The type of regression problem is irrelevant, all we need is a score function such as RSS in linear regression, log-likelihood in generalized regression, partial log-likelihood in Cox regression, or misclassification, that relates fitted values with the response. For simplicity, we assume in here that Y is

Address reprint requests to Dr. Charles Kooperberg, Division of Public Health Sciences, Fred Hutchinson Cancer Research Center, 1100 Fairview Avenue N, MP-1002, Seattle, WA 98109-1024

a binomial random variable. The simplest Logic Regression model is now

$$\hat{Y} = \text{Ind}(L = 1), \tag{1}$$

where L is any logic (Boolean) expression that involves the predictors X_i , such as $L = X_1$ or $L = X_1 \wedge (X_2^c \wedge (X_3 \vee X_4^c))$. Misclassification, $\sum(Y \neq \hat{Y})$, would be the score for equation 1. If we want a regression equation of this form, the main problem is to find good candidates for L , as the collection of all possible logic terms is enormous.

It turns out to be very convenient to write logic expressions in tree form. For example, we can draw $X_1 \wedge (X_2^c \wedge (X_3 \vee X_4^c))$ as the tree in the first panel of Figure 1. Using this “logic tree” representation it is possible to obtain any other logic tree by a finite number of operations such as growing of branches, pruning of branches and changing of leaves (borrowing from CART [Breiman et al. 1984] terminology). In the remaining panels of Figure 1 we show three of the logic trees that can be obtained by applying one operation to the original tree.

Using this representation and these operations on logic trees we can adaptively select L using a (stochastic) simulated annealing algorithm. We start with $L = 1$. Then, at each stage a new tree is selected at random among those that can be obtained by simple operations on the current tree. This new tree always replaces the current tree if it has a better score than the old tree, and it is accepted with a probability that depends on the difference between the scores of the old and the new tree and the stage of the algorithm, otherwise. This simulated annealing algorithm has similarities with the Bayesian CART algorithm [Chipman et al., 1998], in which a CART tree is optimized stochastically. Both of these algorithms are distinct from the *greedy* algorithm employed by CART, in that at any stage they not necessarily pick the move that improves the fit most. Diagnostics, and a scheme that adjust the above-mentioned probabilities slowly enough during this algorithm, guarantee that we will find (close to) the optimal model. An advantage of simulated annealing is that we are much less likely to end up in a local maximum of the scoring function. Properties of the simulated annealing algorithm depend on Markov chain theory, and thus on the set of operations that can be applied to logic trees [Aarts and Korst, 1989].

We should point out that the rules obtained by the Logic Regression algorithm are distinctly different from those found by tree based algorithms, such as CART [Breiman et al., 1984]. For those type of methods, the eventual decision rules are like

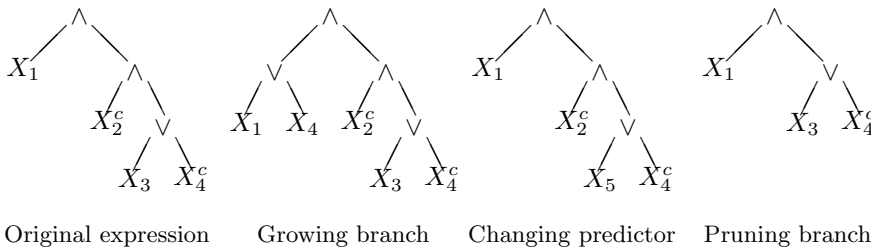


Figure 1: Logic tree representation of $X_1 \wedge (X_2^c \wedge (X_3 \vee X_4^c))$ (first panel) and three logic trees that can be obtained by simple operations on this tree.

$$\text{if } (X_1 \wedge X_2^c \wedge X_3^c) \vee (X_1 \wedge X_2^c \wedge X_4) \text{ predict } \hat{Y} = 1. \quad (2)$$

In general, rules are of the form $\vee_i E_i$, where $E_i = \wedge_j S_{ij}$ and the S_{ij} are simple relations, such as $X_k \in \mathcal{X}$. Rules of this form are said to be in Disjunctive Normal Form (DNF) [Fleisher et al., 1983]. Though any logic expression can be written in DNF, the complexity of such an expression can be reduced considerably if logic expressions of other forms are allowed. Note, for example, that the condition in equation 2 can be reduced to $(X_1 \wedge X_2^c \wedge (X_3^c \vee X_4))$. This latter expression is not in DNF, however.

For complicated problems, we may want to consider more than one logic tree at the same time. Thus, we can extend the classification model (equation 1) (using a binomial likelihood) as

$$\text{logit}(Y = 1|\mathbf{X}) = \beta_0 + \sum_{j=1}^m \beta_j L_j, \quad (3)$$

where each of the L_j is a separate logic tree.

In practice not all predictors may be binary. Continuous predictors can still be included in Logic Regression models by allowing terms like $X_i \leq a$ to enter the model [Ruczinski, 2000]. Alternatively, we can include continuous predictors in a regression model, in addition to logic terms, as we did for the GAW12 data (see Application to the GAW 12 Data).

Using model selection, in addition to a stochastic model building strategy, is of critical importance, as the logic tree with the best score typically overfits the data. A variety of methods of model selection using cross-validation and randomization tests exist [Ruczinski, 2000]. For the GAW12 data we have replicate data; thus we decided to fit our models on one replicate (*training set*), and validate them on another replicate (*test set*).

LOGIC REGRESSION AND GENETIC DATA

In this paper we use the Logic Regression methodology to explore the relationship between single-nucleotide polymorphisms (SNPs) in sequence data and response variables related to a disease outcome. For sequence data we create two predictors for each site with zero, one or two variant alleles. Let $X_1 = 1$ if the site has two variant alleles, and $X_1 = 0$ if it has zero or one variant allele, and let $X_2 = 1$ if the site has one or two variant alleles, and $X_2 = 0$ otherwise. As the selection of a logic tree takes place in an adaptive manner, whether X_1 or X_2 ends up in the logic tree, implies whether a dominant model or recessive model, respectively, for this site best fits the data. As the adaptive methodology removes unnecessary details from the tree, X_1 and X_2 will not end up in the same logic tree, since $X_1 \vee X_2 \equiv X_2$ and $X_1 \wedge X_2 \equiv X_1$, so that the search algorithm automatically reduces such a branch to X_1 or X_2 . When more than one logic tree is fit X_1 can appear in one logic tree, and X_2 can appear in another logic tree, effectively fitting an additive or multiplicative model.

In the application to the GAW12 data we ignore the family structure of the data, and opt for a direct application of the Logic Regression algorithm to a binary and a continuous response. Application of the Logic Regression algorithm to a model that incorporates family data requires identification of a score-function (likelihood)

for a logic model such structure was primarily a matter of convenience. Since we carried out model selection using a test set that satisfies the same family structure as the training set, ignoring the family structure only affects the efficiency of our approach.

APPLICATION TO THE GAW12 DATA

In analyzing the GAW12 data we decided up-front that we would use (part of) the first 25 replicates as training data and the second 25 replicates as an independent test set. Using an independent test-set simplifies the model selection. We applied the Logic Regression algorithm to the 25th replicate data set for the general population. We used the 42nd replicate as the test data set and ignored the family structure. We have sequence data for 1,000 persons. We repeated part of our experiments on a few other replicates and found very similar results. We processed the sequence data for all of the first 25 replicates, keeping only those sites for which among the people that have sequence information fewer than 98% of the persons had zero variant alleles and fewer than 98% had two variant alleles. This left us with 694 sites on the 7 genes combined, that were recoded in $2 \times 694 = 1388$ predictors using the scheme detailed in the previous section.

In the remainder we identify sites and coding of variables as follows: $Gi.D.Sj$ refers to site j on gene i , using dominant coding, i.e. $Gi.D.Sj = 1$ if at least one variant allele exist. Similarly, $Gi.R.Sj$ refers to site j on gene i , using recessive coding, i.e. $Gi.R.Sj = 1$ if two variant alleles exist. We identify complements by the superscript c , e.g. $Gi.D.Sj^c$.

Affected status. As our primary response variables we used the affected status. We fitted a logistic regression model of the form

$$\text{logit}(\text{affected}) = \beta_0 + \beta_1 \times \text{environ}_1 + \beta_2 \times \text{environ}_2 + \beta_3 \times \text{gender} + \sum_{i=1}^K \beta_{i+3} \times L_i. \quad (4)$$

Here gender was coded as 1 for female and 0 for male, environ_j , $j = 1, 2$, are the two environmental factors that were provided, and the L_i , $i = 1, \dots, K$ are logic expressions based on the 1,388 predictors that were created from the sequence data.

Initially we fit models with $K = 1, 2, 3$, allowing logic expressions of at most size 8 on the training data. In Figure 2 we show the deviance of the various fitted Logic

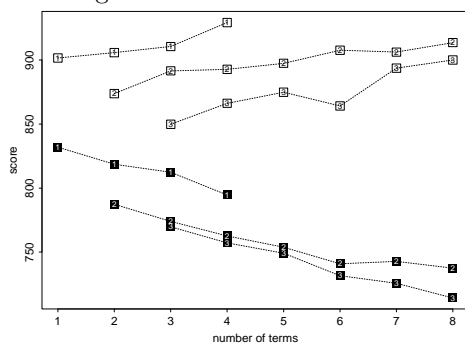


Figure 2. Training (solid) and test (open) set deviances for Logic Regression models for the affected state. The number in the boxes indicate the number of logic trees.

Regression models. As very likely the larger models overfit the data, we validated the models by computing the fitted deviance for an independent test set keeping the models fixed at those selected. These results are also shown in Figure 2. From this figure we see that the models with three logic trees with a total of three and six leaves have the lowest test-set deviance. As the goal of the current investigation is to identify sites that are possibly linked to the outcome, we prefer the larger of these two models. In addition, when we repeated the experiment on a training set of five replicates and a test set of 25 replicates the model with six leaves had a slightly lower test set deviance than the model with three leaves. We carried out a randomization test, conditioning on the model with three leaves, to determine how much better a model with six leaves fits the data if a model with three leaves is the true model. As the improvement that we observed is much larger than what would be expected by chance, this confirmed that the model with six leaves fits the data better than a model with three leaves. The model with six leaves that was fitted on the single replicate is presented in Figure 3. The logistic regression model corresponding to this Logic Regression model is

$$\text{logit}(\text{affected}) = 0.44 + 0.005 \times \text{environ}_1 - 0.27 \times \text{environ}_2 + 1.98 \times \text{gender} \\ - 2.09 \times L_1 + 1.00 \times L_2 - 2.82 \times L_3.$$

All but the second environment variable in this model are statistically significant.

Note that for all 1000 persons with sequence data in replicate 25 site 76 on gene 1 is exactly the opposite of site 557, which was indicated as the correct site on the solutions (for example, a person with v variant alleles on site 76 always has $2 - v$ variant alleles on site 557). Similarly, the Logic Regression algorithm identified site 5007 on gene 6, which is identical for all 1,000 persons to site 5782, the site which was indicated on the solutions. We note that the “correct” site on gene 1 appears twice in the logic tree model. Once, as a recessive coding (*G1.R.S557*) and one effectively as a dominant coding (*G1.R.S76^c ≡ G1.D.S557* on this replicate) for site 557, suggesting that the true model may have been additive. When two sites are in (almost) perfect disequilibrium, as is the case for these sites, the Logic Regression algorithm may identify one of these sites as the algorithm cannot distinguish between them. The three remaining leaves in the model are all part of gene 2: two site close to the ends of the gene and one site in the center.

Quantitative trait 5. In the solutions to the GAW12 data it was described that Quantitative trait 5 (Q5) depended on the sequence data of gene 2, but the exact pattern of the mutations that were influencing this trait were not given. To investigate this further we decided to fit another Logic Regression model of the form equation 4, with Q5 as the response variable using linear regression. We would

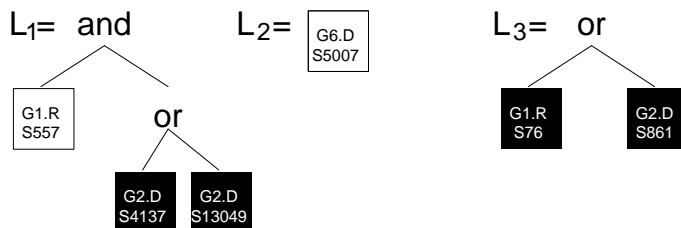


Figure 3. Fitted Logic Regression model for the affected state data with three trees and six leaves. Variables that are printed white on a black background are the complement of those indicated.

expect that this way we would be better able to find a more precise dependence on gene 2 than for the logistic model using affected status as the response. We carried out model selection identical to that for the logistic regression described above. While the solutions indicated that the (sequence) dependence of Q5 was only on gene 2, we allowed all genes in the model. The model that was selected had three logic trees and a total of seven leaves. The three trees were

$$L_1 = (G2.D.S851 \wedge G2.D.S1289^c) \vee G2.R.S8657^c,$$

and

$$L_2 = G2.D.S1400 \vee G2.D.S4977,$$

$$L_3 = G2.D.S334 \vee G2.D.S10091.$$

Thus, a model that depends on a large number of sites on gene 2 is fit. The solutions indicate that Quantitative trait 5 indeed depends on gene 2 and not on the other genes. (Models with different numbers of trees or leaves for Q5 always exclusively depended on sites on gene 2.) Except for site 8657, all sites occurred in the fitted model as dominant genes. While this could be related to the way the data was generated, it is also possible that the data was generated using an additive model, as for six of the seven sites that were selected few people had two variants, and the power of selecting recessive codings of a site is thus smaller than that of selecting the dominant codings of the same site.

DISCUSSION

Our analysis of the GAW12 data shows the potential usefulness of Logic Regression. While our algorithm was provided with data on hundreds of predictors (sites), it correctly picked out those few that were the responsible sites in the underlying model. No tweaking of the algorithm was needed to achieve these results.

In applying Logic Regression, it is advantageous to use raw sequence data, rather than data that has been aggregated as haplotypes, as such predictors would yield few categorical variables with many levels, while sequence data yields many variables with few levels. In combining these variables the Logic Regression algorithm, effectively, determines which levels of the haplotype are associated with disease.

ACKNOWLEDGEMENTS

We thank Sue Li for helpful discussions. Research supported in part by NIH grant CA 74841.

REFERENCES

- Aarts EHL, Korst JHM. 1989. Simulated annealing and Boltzmann machines. New York: Wiley.
- Breiman L, Friedman JH, Olshen RA, et al. 1984. Classification and Regression Trees. Belmont, California: Wadsworth.
- Chipman H, George E, McCulloch R. 1998. Bayesian CART model search (with discussion). J. Am. Statist. Assoc. 93:935-60.
- Fleisher H, Tavel M, and Yeager Y. 1983. Exclusive-or representation of boolean functions. IBM J. Res. Develop. 27:412-16.
- Ruczinzki I. 2000. Logic Regression and statistical issues related to the protein folding problem. Ph. D. thesis. Seattle: University of Washington, Dept. of Statistics.
- Ruczinzki I, Kooperberg C, LeBlanc ML. 2001. Logic Regression technical report. Seattle: Fred Hutchinson Cancer Research Center.